AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions, and listings, of claims in the application:

1. (Original) Apparatus for processing data, said apparatus comprising:

(i) a processor core operable to execute native instructions of a native instruction set; and

(ii) an instruction translator operable to interpret non-native instructions of a non-native instruction set into native instructions for execution by said processor core; wherein

(iii) said instruction translator is responsive to a return to non-native instruction of said non-native instruction set to return processing to a non-native instruction; and

(iv) said instruction translator is responsive to a return to native instruction of said non-native instruction set to return processing to a native instruction.

2. (Original) Apparatus as claimed in claim 1, wherein said instruction translator is a hardware based instruction translator.

3. (Original) Apparatus as claimed in claim 1, wherein said instruction translator is a software based interpreter.

4. (Original) Apparatus as claimed in claim 1, wherein said instruction translator is a combination of a hardware based instruction translator and a software based interpreter.

5. (Previously Presented) Apparatus as claimed in claim 1, wherein said non-native instructions are Java Virtual Machine instructions.

6. (Previously Presented) Apparatus as claimed in claim 1, wherein a non-native subroutine is called from native code via a non-native veneer subroutine, such that, upon completion of said non-native subroutine, a return to non-native instruction can be used to return processing to said non-native veneer subroutine with a return to native instruction within said non-native veneer subroutine serving to return processing to said native code.

7. (Original) Apparatus as claimed in claim 6, wherein said non-native subroutine is also called from non-native code.

8. (Previously Presented) Apparatus as claimed in claim 6, wherein said non-native veneer subroutine is dynamically created when said non-native subroutine is called from native code.

9. (Original) Apparatus as claimed in claim 8, wherein said non-native veneer subroutine is created stored within a stack memory area used by native code operation.

10. (Previously Presented) Apparatus as claimed in claim 1, wherein said instruction translator is responsive to a plurality of types of return to non-native instruction.

11. (Original) Apparatus as claimed in claim 10, wherein said plurality of types of return to non-native instruction are operable to return with respective different types of return value.

12. (Original) Apparatus as claimed in claim 11, wherein said plurality of different types of return value include one or more of:

(i) a 32-bit integer return value;

(ii) a 64-bit integer return value;

(iii) an object reference return value;

(iv) a single precision floating point return value;

(v) a double precision floating point return value; and

(vi) a void return value having no value.

13. (Previously Presented) Apparatus as claimed in claim 1, wherein said instruction translator is responsive to a plurality of types of return to native instruction.

14. (Original) Apparatus as claimed in claim 13, wherein said plurality of types of return to native instruction are operable to return with respective different types of return value.

15. (Original) A method of processing data, said method comprising the steps of:

(i) executing native instructions of a native instruction set using a processor core; and

(ii) interpreting non-native instructions of a non-native instruction set into native instructions for execution by said processor core; wherein

(iii) in response to a return to non-native instruction of said non-native instruction set, returning processing to a non-native instruction; and

(iv) in response to a return to native instruction of said non-native instruction set, returning processing to a native instruction.

16. (Original) A computer program product carrying a computer program for controlling a data processing apparatus in accordance with the method of claim 15.

17. (Canceled).

18. (Canceled).

19. (Canceled).

20. (New) Apparatus for processing data, comprising:

processing means for executing native instructions of a native instruction set; and

translator means for interpreting non-native instructions of a non-native instruction set into native instructions for execution by said processor core being responsive to a return to non-native instruction of said non-native instruction set to return processing to a non-native instruction, and responsive to a return to native instruction of said non-native instruction set to return processing to a native instruction.

21. (New) Apparatus as claimed in claim 1, further comprising:

means for calling a non-native subroutine from native code via a non-native veneer subroutine, such that, upon completion of said non-native subroutine, a return to non-native instruction can be used to return processing to said non-native veneer subroutine with a return to native instruction within said non-native veneer subroutine serving to return processing to said native code.

22. (New) Apparatus as claimed in claim 21, further comprising:

means for calling said non-native subroutine from non-native code.

23. (New) Apparatus as claimed in claim 21, further comprising:

means for dynamically creating said non-native veneer subroutine when said non-native subroutine is called from native code.

24. (New) Apparatus as claimed in claim 23, further comprising:

means for storing said created non-native veneer subroutine within a stack memory area used by native code operation.

25. (New) Apparatus as claimed in claim 20, wherein said translator means is responsive to a plurality of types of return to non-native instruction.

26. (New) Apparatus as claimed in claim 25, wherein said plurality of types of return to non-native instruction are operable to return with respective different types of return value.

27. (New) Apparatus as claimed in claim 20, wherein said instruction translator is responsive to a plurality of types of return to native instruction.

28. (New) Apparatus as claimed in claim 27, wherein said plurality of types of return to native instruction are operable to return with respective different types of return value.